AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1   1.   (Currently Amended) In a communication network, a method of TCP state
2   migration comprising the steps of:
3   a)   establishing a TCP/IP communication session between a client computer
4   and a first server computer, said first server computer part of a plurality of server
5   computers forming a web cluster containing information, said communication session
6   established for the transfer of data contained within said information;
7   b)   handing off said communication session to a selected server computer
8   from said first server computer over a persistent control channel using TCP handoff
9   modules that are dynamically loadable within TCP/IP stacks in operating systems located
10   at both said first server computer and said selected server computer, ~~that~~ wherein the TCP
11   handoff modules implement a TCP handoff protocol that works within kernel levels of ~~an~~
12   ~~existing TCP/IP protocol~~the operating systems; and
13   c)   migrating a first TCP state of said first server computer to said selected
14   server computer, and a second TCP state of said selected server computer to said first
15   server computer over said control channel.

1   2.   (Currently Amended) The method as described in Claim 1, wherein said

2   step a) comprises the steps of:

3   receiving a SYN packet from said client computer at a first ~~BTCP~~ bottom TCP

4   (BTCP) module located at said first server computer;

5   sending said SYN packet upstream to a first TCP module located above said first

6   BTCP module in a first operating system of said first server computer;

7   receiving a first SYN/ACK packet from said first TCP module;

8   parsing said first ~~initial~~ TCP state from said first SYN/ACK packet, including a

9   first initial sequence number for said first TCP module associated with said TCP/IP

10   communication session;

11   sending said SYN/ACK packet to said client computer;

12   receiving an ACK packet from said client at said first BTCP module;

13   sending said ACK packet to said first TCP module;

14   receiving a web request packet associated with said TCP/IP communication

15   session at said first BTCP module at said first server computer;

16   storing said SYN packet, said ACK packet and said web request packet at said

17   first server computer.

1   3.   (Currently Amended) The method as described in Claim 2, wherein said

2   step b) comprises the steps of:

3   examining content of said web request packet;

4   determining which of said plurality of server computers, a selected server

5   computer, can best process said ~~WEB~~ web request packet, based on said content, wherein

6   the server computer determined to be able to best process said web request packet is the

7   selected server computer;

8   sending a handoff request packet from said first BTCP module to a second BTCP

9   module at said selected server computer over said control channel, if said selected server

10   computer is not said first server computer;

11   including said SYN packet and said ACK packet in said handoff request packet;

12   changing a ~~first~~ destination IP address of said SYN packet to a second IP address

13   of said selected server computer, at said second BTCP module;

14   sending said SYN packet from said second BTCP module to ~~said~~ a second TCP

15   module at said selected server computer;

16   receiving a second SYN/ACK packet at said second BTCP module;

17   parsing said second ~~initial~~ TCP state from said second SYN/ACK packet,

18   including a second initial sequence number, for said second TCP module, that is

19   associated with said TCP/IP communication session;

20   changing a ~~second~~ destination IP address of said ACK packet to said second IP

21   address, at said second BTCP module;

22   updating said ACK packet to reflect said second TCP state of said selected server

23   computer in said communication session;

24   sending said ACK packet that is updated from said second BTCP module to said

25   second TCP module; and

26   sending a handoff acknowledgment message from said second BTCP module to

27   said first BTCP module.

1   4.   (Currently Amended) The method as described in Claim 3, wherein step c)

2   comprises ~~the steps of~~:

3   ~~monitoring traffic associated with establishing said TCP/IP communication~~

4   ~~session in step a), at said first BTCP module, to parse a first initial TCP state of said first~~

5   ~~server computer, said first initial TCP state associated with said TCP/IP communication~~

6   ~~session; and~~

7   migrating said first ~~initial~~ TCP state to said second BTCP module over said

8   control channel by including said first initial ~~TCP state~~ sequence number in said handoff

9   request packet, ~~said first initial TCP state including a first sequence number,~~ such that

10   said second BTCP module can calculate said first TCP state for said first server computer

11   in said TCP/IP communication session.

1   5.   (Currently Amended) The method as described in Claim 3, wherein step c)

2   comprises ~~the steps of~~:

3   ~~monitoring traffic associated with handing off said TCP/IP communication~~

4   ~~session at said second BTCP module, to parse a second initial TCP state of said selected~~

5   ~~server computer, said second initial TCP state associated with said TCP/IP~~

6   ~~communication session; and~~

7   migrating said second ~~initial~~ TCP state of said selected server computer to said

8   first BTCP module by including said second initial ~~TCP state~~ sequence number in said

9   handoff acknowledgment ~~packet~~message, ~~said second initial TCP state including a~~

10   ~~second initial sequence number,~~ such that said first BTCP module can calculate said

11   second TCP state for said selected server computer in said TCP/IP communication

12   session.

1   6.   (Currently Amended) The method as described in Claim 2, comprising the

2   further steps of:

3   intercepting a connection indication message sent from said first TCP module to

4   an application layer above said first TCP module at a first upper-TCP (UTCP) module at

5   the first server computer, said connection indication message sent by said first TCP

6   module upon establishing said communication session; and

7   holding said connection indication message at said first UTCP module, wherein

8   said first UTCP module and said first BTCP module provide a wrapper around said first

9   TCP module.

1   7.   (Currently Amended) The method as described in Claim 6, wherein said

2   method comprises the further steps of:

3   sending a reset packet from said first BTCP module upon receiving said handoff

4   acknowledgment packet message to said first TCP module;

5   discarding said connection indication message at said first UTCP module;

6   receiving incoming data packets from said client at said first BTCP module;

7   changing said destination addresses of said incoming data packets to said second

8   IP address;

9   updating sequence numbers and TCP checksum in said data packets to reflect said

10 second TCP state of said selected server computer; and

11   forwarding said updated data packets to said selected server computer.

1   8.   (Currently Amended) The method as described in Claim 6, comprising the

2   further steps of:

3   sending notification from said first BTCP module to said first UTCP module to

4   release said connection indication message, if said selected server computer is said first

5   server computer;

6   sending incoming data packets, including said web request packet, from said

7   client computer, received at said first BTCP module, upstream.

1   9.   (Currently Amended) The method as described in Claim [[1]]2,

2   comprising the further ~~step~~steps of:

3   intercepting outgoing response packets from said selected server computer at a

4   second bottom TCP (BTCP) module located at said selected server computer;

5   changing, by the second BTCP module, source addresses of said response packets

6   to a first IP address of said first server computer;

7   updating, by the second BTCP module, sequence numbers and TCP checksum in

8   said response packets to reflect said first TCP state of said first server computer; and

9   sending said updated response packets to said client computer without passing the

10   updated response packets through the first server computer.

1   10.   (Currently Amended) The method as described in Claim [[1]]9,

2   comprising the further steps of:

3   monitoring TCP/IP control traffic for said communication session at said second

4   BTCP module;

5   understanding when said communication session is closed at said ~~second~~ selected

6   server computer;

7   sending a termination message to said first server computer over said control

8   channel;

9   terminating said TCP/IP communication session at said first server computer by

10   terminating a forwarding mode at said first BTCP module; and

11   freeing data resources associated with said communication session at said first

12   server computer.

1   11.   (Currently Amended) In a communication network, a method of TCP state

2   migration comprising the steps of:

3   a)   establishing a TCP/IP communication session between a client computer

4   and a first server computer, said first server computer part of a plurality of server

5   computers forming a web cluster containing information, said communication session

6   established for the transfer of data contained within said information;

7   b)   monitoring traffic associated with establishing said TCP/IP

8   communication session to understand a first initial TCP state of said first server computer

9   associated with said TCP/IP communication session, at a first bottom-TCP (BTCP)

10   module at said first server computer, wherein said first BTCP module is a dynamically

11   loadable kernel module (DLKM) loaded in said first server computer without modifying

12   a first operating system of said first server computer;

13   c)   receiving a web request associated with said TCP/IP communication

14   session at said first BTCP module at said first server computer;

15   d)   examining content of said web request;

16   e)   determining which of said plurality of server computers, a selected server

17   computer, can best process said web request, based on said content;

18   f)   handing off said communication session to said selected server computer

19   from said first server computer over a persistent control channel, if said selected server

20   computer is not said first server computer;

21   g)   monitoring traffic associated with handing off said TCP/IP communication

22   session to understand a second initial TCP state of said selected server computer

23   associated with said TCP/IP communication session, at a second BTCP module at said

24   selected server computer, wherein said second BTCP module is a DLKM loaded in said

25   selected server computer without modifying a second operating system of said selected

26   server computer;

27   h)   migrating said first initial TCP state to said selected server computer over

28   said control channel, such that said second BTCP module can calculate a first TCP state

29   for said first server computer in said TCP/IP communication session;

30   i)   sending a second initial TCP state of said selected server computer to said

31   first BTCP module, such that said first BTCP module can calculate a second TCP state

32   for said selected server computer in said TCP/IP communication session;

33   j)   forwarding data packets received at said first BTCP module from said

34   client to said selected server computer, by changing said data packets to reflect said

35   second TCP state and a second IP address of said selected server computer;

36   k)   sending response packets from said selected server computer directly to

37   said client computer by changing said response packets to reflect said first TCP state and

38   a first IP address of said first server computer; and

39   l)   terminating said TCP/IP communication session at said first server

40   computer when said TCP/IP communication session is closed.

1   12.   (Currently Amended) The method as described in Claim 11, wherein said

2   ~~step~~steps a) and b) comprise~~comprises~~ the steps of:

3   receiving a SYN packet from said client computer at said first BTCP module;

4   sending said SYN packet upstream to a first TCP module located above said first

5   BTCP module in [[a]]said first operating system of said first server computer;

6   receiving a first SYN/ACK packet from said first TCP module;

7   parsing, by the first BTCP module, said first initial TCP state from said first

8   SYN/ACK packet, including a first initial sequence number for said first TCP module

9   associated with[[,]] said TCP/IP communication session;

10   sending said SYN/ACK packet to said client computer;

11   receiving, by the first BTCP module, an ACK packet from said client computer at

12   said first BTCP module;

13   sending said ACK packet to said first TCP module;

14   storing said SYN packet, said ACK packet and said web request at said first

15   server computer.

1   13.   (Currently Amended) The method as described in Claim [[11]]12, wherein

2 said ~~step e)~~steps f) and g) comprise~~comprises~~ the steps of:

3   sending a handoff request packet from said first BTCP module to said second

4 BTCP module over said control channel;

5   including said SYN packet and said ACK packet in said handoff request packet;

6   changing a ~~first~~ destination IP address of said SYN packet to a second IP address

7 of said selected server computer, at said second BTCP module;

8   sending, by said second BTCP module, said SYN packet to ~~said~~ a second TCP

9 module at said selected server computer;

10   receiving a second SYN/ACK packet at said second BTCP module;

11   parsing said second initial TCP state from said second SYN/ACK packet,

12 including a second initial sequence number, for said second TCP module, that is

13 associated with said TCP/IP communication session;

14   changing a ~~second~~ destination IP address of said ACK packet to said second IP

15 address, at said second BTCP module;

16   updating said ACK packet to reflect said second TCP state of said selected server

17 computer in said communication session;

18   sending said ACK packet that is updated from said second BTCP module to said

19 second TCP module; and

20   sending a handoff acknowledgment message from said second BTCP module to

21 said first BTCP module.

1   14.   (Currently Amended) The method as described in Claim 13, wherein said

2 ACK packet includes said first initial TCP state of said first server computer ~~as provided~~

3 ~~for in step f)~~.

10

1   15.   (Currently Amended) The method as described in Claim 13, wherein said

2   handoff acknowledgment <u>message </u>includes said ~~second initial TCP state of said second~~

3   ~~server computer, including a~~ second initial sequence number~~, for said second TCP~~

4   ~~module, that is associated with said TCP/IP communication session as provided for in~~

5   ~~step i)~~.

1   16.   (Currently Amended) The method as described in Claim 13, comprising

2   the further steps of:

3   intercepting a connection indication message sent from said first TCP module to

4   an application layer above said first TCP module at a first upper-TCP (UTCP) module<u> at</u>

5   <u>said first server computer</u>, said connection indication message sent by said first TCP

6   module upon establishing said communication session; and

7   holding said connection indication message at said first UTCP module<u>, wherein</u>

8   <u>said first UTCP module and said first BTCP module provide a wrapper around said first</u>

9   <u>TCP module</u>.

1   17.   (Currently Amended) The method as described in Claim 16, ~~wherein step~~

2   ~~h) comprises~~<u>comprising </u>the further steps of:

3   sending a reset packet from said first BTCP module upon receiving said handoff

4   acknowledgment ~~packet~~<u>message </u>to said first TCP module;

5   discarding said connection indication message at said first UTCP module;

6   receiving incoming data packets from said client at said first BTCP module;

7   changing ~~said ~~destination addresses of said incoming data packets to said second

8   IP address;

9   updating sequence numbers and TCP checksum in said data packets to reflect said

10   second TCP state of said selected server computer; and

11   forwarding said <u>updated </u>data packets to said selected server computer.

1   18.   (Currently Amended) The method as described in Claim 11, wherein step

2   k) comprises the steps of:

3   intercepting outgoing response packets from said selected server computer at said

4   second BTCP module;

5   changing, by said second BTCP module, source addresses of said response

6   packets to said first IP address;

7   updating, by said second BTCP module, sequence numbers and TCP checksum in

8   said response packets to reflect said first TCP state of said first server computer; and

9   sending said updated response packets to said client computer without passing the

10 updated response packets through said first server computer.

1   19.   (Currently Amended) The method as described in Claim 11, wherein step

2   1) comprises the steps of:

3   monitoring TCP/IP control traffic for said communication session at said second

4   BTCP module;

5   understanding when said communication session is closed at said second selected

6   server computer;

7   sending a termination message to said first server computer over said control

8   channel;

9   terminating a forwarding mode at said first BTCP module; and

10   freeing data resources associated with said communication session at said first

11 server computer.

1   20.   (Currently Amended) The method as described in Claim 16, comprising

2   the further steps of:

3   sending notification from said first BTCP module to said first UTCP module to

4   release said connection indication message, if said selected server computer is said first

5   server computer; and

6   sending incoming data packets, including said web request, from said client

7   computer, received at said first BTCP module, upstream.

12

1   21.   (Currently Amended) The method as described in Claim 11, wherein each

2   of said plurality of server computers is constructed similarly including BTCP modules

3   located downstream from respective TCP modules in the corresponding server

4   computers, and UTCP modules located upstream from the corresponding TCP modules,

5   wherein each pair of the BTCP modules and UTCP modules provides a wrapper around a

6   corresponding TCP module.

1   22.   (Cancelled)

1   23.   (Currently Amended) The method as described in Claim [[22]]21, wherein

2   said control channel allows for communication between all UTCP modules of the

3   corresponding server computers.

1   24.   (Original) The method as described in Claim 11, wherein said plurality of

2   server computers is coupled together over a wide area network in said communication

3   network.

1   25.   (Original) The method as described in Claim 11, wherein said information

2   is partitioned/partially replicated throughout each of said plurality of server computers.

1   26.   (Currently Amended) A <u>first</u> server computer comprising:

2   <u>an operating system;</u>

3   <u>a TCP module in the operating system;</u>

4   ~~an~~ <u>a first</u> upper TCP (UTCP) module located above [[a]]<u>the</u> TCP module in

5   [[an]]<u>the</u> operating system of said <u>first</u> server computer;

6   a <u>first</u> bottom TCP (BTCP) module located below said TCP module, <u>wherein</u> said

7   <u>first</u> UTCP, TCP, and <u>first</u> BTCP modules <u>implement</u> ~~implementing a method of~~ handing

8   off a communication session between <u>the first server computer and a second node,</u>

9   <u>wherein the first BTCP and first UTCP modules are dynamically loadable kernel modules</u>

10   <u>(DLKMs) that are dynamically loadable and unloadable in said first server computer</u>

11   <u>without modifying the operating system</u>~~a first node and second node in a cluster network~~

12   ~~that works within the kernel level of an existing TCP/IP protocol, by migrating TCP~~

13   ~~states associated with said first and second node~~s.

1   27.   (Currently Amended) The first server computer as described in Claim 26,

2 wherein said method comprises the steps of:

3   a)   establishing a TCP/IP communication session is established between a

4 client computer and said server computer, said first node, said first server computer part

5 of a plurality of server computers including said second node forming said a cluster that

6 contains network containing information, said communication session established for the

7 transfer of data contained within said information;

8   b)   receiving wherein said first BTCP module is configured to:

9   receive a web request associated with said TCP/IP communication session

10 at a first BTCP module at said server computer;

11   c)   examining examine content of said web request;

12   d)   determining determine which of said plurality of server computers,

13 a selected server computer, can best process said web request, based on said content,

14 wherein the server computer determined to be able to best process said web request is the

15 second node;

16   e)   handing hand off said communication session to said selected

17 server computer second node from said first server computer over a persistent control

18 channel, if said selected server computer second node is not said first server computer,

19 wherein the handing off causes migration of a first TCP state of said first server computer

20 to the second node, and migration of a second TCP state of said second node to the server

21 computer; and

22   f)   migrating a first TCP state of said server computer to said selected server

23 computer, and sending a second TCP state of said selected server computer to said server

24 computer over said control channel.

1   28.   (Currently Amended) The <u>first</u> server computer as described in Claim 27,

2   wherein <u>said first BTCP module is configured to further</u>~~step a) of said method comprises~~

3   ~~the steps of~~:

4   ~~receiving~~<u>receive</u> a SYN packet from said client <u>computer</u>~~at said BTCP module~~;

5   ~~sending~~<u>send</u> said SYN packet upstream to said TCP module;

6   ~~receiving~~<u>receive</u> a first SYN/ACK packet from said TCP module;

7   ~~parsing~~<u>parse</u> a first initial TCP state from said first SYN/ACK packet, including a

8   first initial sequence number for said TCP module associated with said TCP/IP

9   communication session;

10   ~~sending~~<u>send</u> said SYN/ACK packet to said client<u> computer</u>;

11   ~~receiving~~<u>receive</u> an ACK packet from said client<u> computer</u>~~at said BTCP module~~;

12   ~~sending~~<u>send</u> said ACK packet to said TCP module~~;~~

13   ~~storing said SYN, ACK at said server computer~~.

1   29.   (Currently Amended) The <u>first</u> server computer as described in Claim 28,

2   wherein said <u>first BTCP module is configured to further</u>~~method comprises the steps of~~:

3   ~~sending~~<u>send</u> a handoff request packet ~~from said BTCP module~~to a second BTCP

4   module over said control channel, said second BTCP module located below a second

5   TCP module in a second operating system at said <u>second node</u>~~selected server computer~~;

6   ~~including~~<u>include</u> said SYN packet and said ACK packet in said handoff request

7   <u>packet</u>;

8   ~~receiving~~<u>receive</u> a handoff acknowledgment message ~~at said BTCP module~~from

9   said second BTCP module.

1   30.   (Cancelled)

16

1   31.   (Currently Amended) The <u>first</u> server computer as described in Claim 29,

2   wherein said <u>first UTCP module is configured to</u>method comprises the further steps of:

3   intercepting <u>intercept</u> a connection indication message sent from said first TCP

4   module to an application layer above said first TCP module at a first upper TCP (UTCP)

5   module, said connection indication message sent by said first TCP module upon

6   establishing said communication session; and

7   holding <u>hold</u> said connection indication message at said first UTCP module.

1   32.   (Currently Amended) The computer <u>first server</u> system as described in

2   Claim 31, wherein said <u>first BTCP module is configured to further</u>method comprises the

3   further steps of:

4   sending <u>send</u> a reset packet from said first BTCP module upon receiving said

5   handoff acknowledgment <u>message</u> packet to said first TCP module;

6   discarding said connection indication message at said first UTCP module;

7   receiving <u>receive</u> incoming data packets from said client <u>computer</u> at said first

8   BTCP module;

9   changing said <u>change</u> destination addresses of said incoming data packets to said

10   <u>a</u> second IP address <u>of said second node</u>;

11   updating <u>update</u> sequence numbers and TCP checksum in said data packets to

12   reflect said second TCP state of said <u>second node</u>selected server computer; and

13   forwarding <u>forward</u> said <u>updated</u> data packets to said <u>second node</u>selected server

14   computer.

1   33.   (Currently Amended) The <u>first</u> server computer as described in Claim 31,

2   <u>wherein said first BTCP module is configured to further</u>said method comprising the

3   further steps of:

4   sending <u>send</u> notification from said BTCP module to said <u>first</u> UTCP module to

5   release said connection indication message, if said <u>second node</u>selected server computer

6   is said <u>first</u> server computer;

7   sending <u>send</u> incoming data packets, including said web request, from said client

8   <u>computer</u>, received at said first BTCP module, upstream.

1   34.   (Currently Amended) The <u>first</u> server computer as described in Claim 26,

2 wherein said first BTCP module is configured to~~said method comprising the further steps~~

3 ~~of~~:

4   ~~receiving~~<u>receive</u> a handoff request from a ~~first~~<u>second</u> BTCP module located at a

5 ~~first~~<u>second</u> server computer ~~within said cluster network~~ over a persistent control channel,

6 said ~~first~~<u>second</u> server computer having established a <u>second</u> communication session

7 with a <u>second</u> client computer, ~~said communication session established for the transfer of~~

8 ~~data contained within said server computer,~~ said handoff request including a SYN packet

9 and an ACK packet, said SYN<u> packet</u> and ACK packet used for establishing said <u>second</u>

10 communication session between said <u>second</u> client <u>computer</u> and said ~~first~~<u>second</u> server

11 computer, said ACK packet including a first initial TCP state <u>including a first initial TCP</u>

12 <u>sequence number</u> of said ~~first~~<u>second</u> server computer in said communication session~~,~~

13 ~~including a first initial TCP sequence number~~;

14   ~~changing~~<u>change</u> a ~~first~~ destination IP address of said SYN packet to a ~~second~~<u>first</u>

15 IP address of said <u>first</u> server computer~~, at said BTCP module~~;

16   ~~sending~~<u>send</u> said SYN packet to said TCP module;

17   ~~receiving~~<u>receive</u> a SYN/ACK packet ~~at said second BTCP module~~;

18   ~~parsing~~<u>parse</u> a second initial TCP state from ~~second~~<u>said</u> SYN/ACK packet,

19 including a second initial sequence number, for said TCP module~~, said second initial TCP~~

20 ~~state associated with a second TCP state for said server computer in said TCP/IP~~

21 ~~communication session~~;

22   ~~changing~~<u>change</u> a ~~second~~ destination IP address of said ACK packet to said

23 ~~second~~<u>first</u> IP address~~, at said BTCP module~~;

24   ~~updating~~<u>update</u> said ACK packet to reflect said second TCP state ~~of said selected~~

25 ~~server computer in said communication session~~;

26   ~~sending~~<u>send</u> said ACK packet that is updated to said TCP module; and

27   ~~sending~~<u>send</u> a handoff acknowledgment message to said ~~first~~<u>second</u> BTCP

28 module over said control channel.

1   35.-37. (Cancelled)

1   38.   (New) The method as described in Claim 1, wherein the TCP handoff

2 modules are dynamically loadable kernel modules, the method further comprising:

3   dynamically loading the TCP handoff modules in the corresponding first server

4 computer and selected server computer without modifying the operating systems of the

5 respective first server computer and selected server computer.

1   39.   (New) The method as described in claim 6, wherein the first BTCP

2 module and first UTCP module are dynamically loadable kernel modules.